PCT WORLD INTELLECTUAL PROPERTY ORGANIZATION
International Bureau

INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification 6 : H03K 19/177 — A1

(11) International Publication Number: WO 95/04404

(43) International Publication Date: 9 February 1995 (09.02.95)

(21) International Application Number: PCT/US94/07187

(22) International Filing Date: 24 June 1994 (24.06.94)

(30) Priority Data:
08/101,197 3 August 1993 (03.08.93) US

(71) Applicant: ADVANTAGE LOGIC, INC. [US/US]; Suite 456, 20863 Stevens Creek Boulevard, Cupertino, CA 95014 (US).

(72) Inventor: TING, Benjamin, S.; 21120 Sullivan Way, Saratoga, CA 95070 (US).

(74) Agents: HAO, James, P. et al.; Blakely, Sokoloff, Taylor & Zafman, 7th floor, 12400 Wilshire Boulevard, Los Angeles, CA 90026 (US).

(81) Designated States: AM, AT, AU, BB, BG, BR, BY, CA, CH, CN, CZ, DE, DK, ES, FI, GB, GE, HU, JP, KE, KG, KP, KR, KZ, LK, LU, LV, MD, MG, MN, MW, NL, NO, NZ, PL, PT, RO, RU, SD, SE, SI, SK, TJ, TT, UA, UZ, VN, European patent (AT, BE, CH, DE, DK, ES, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, ML, MR, NE, SN, TD, TG).

Published
With international search report.

(54) Title: ARCHITECTURE AND INTERCONNECT SCHEME FOR PROGRAMMABLE LOGIC CIRCUITS

(57) Abstract

An architecture and distributed hierarchical interconnect scheme for field programmable gate arrays (FPGAs). The FPGA is comprised of a number of cells which perform logical functions on input signals. Programmable intraconnections provide connectability between each output of a cell belonging to a logical cluster to at least one input of each of the other cells belonging to that logical cluster. A set of programmable block connectors are used to provide connectability between logical clusters of cells and accessibility to the hierarchical routing network. A uniformly distributed first layer of routing network lines is used to provide connections amongst sets of block connectors. A uniformly distributed second layer of routing network lines is implemented to provide connectability between different first layers of routing network lines. Switching networks are used to provide connectability between the block connectors and routing network lines corresponding to the first layer. Other switching networks provide connectability between the routing network lines corresponding to the first layer to routing network lines corresponding to the second layer. Additional uniformly distributed layers of routing network lines are implemented to provide connectability between different prior layers of routing network lines. An additional routing layer is added when the number of cells is increased as a square function of two of the prior cell count in the array while the length of the routing lines and the number of routing lines increases as a linear function of two. Programmable bi-directional passgates are used as switches to control which of the routing network lines are to be connected.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	**GB**	United Kingdom	**MR**	Mauritania
AU	Australia	**GE**	Georgia	**MW**	Malawi
BB	Barbados	**GN**	Guinea	**NE**	Niger
BE	Belgium	**GR**	Greece	**NL**	Netherlands
BF	Burkina Faso	**HU**	Hungary	**NO**	Norway
BG	Bulgaria	**IE**	Ireland	**NZ**	New Zealand
BJ	Benin	**IT**	Italy	**PL**	Poland
BR	Brazil	**JP**	Japan	**PT**	Portugal
BY	Belarus	**KE**	Kenya	**RO**	Romania
CA	Canada	**KG**	Kyrgystan	**RU**	Russian Federation
CF	Central African Republic	**KP**	Democratic People's Republic of Korea	**SD**	Sudan
CG	Congo			**SE**	Sweden
CH	Switzerland	**KR**	Republic of Korea	**SI**	Slovenia
CI	Côte d'Ivoire	**KZ**	Kazakhstan	**SK**	Slovakia
CM	Cameroon	**LI**	Liechtenstein	**SN**	Senegal
CN	China	**LK**	Sri Lanka	**TD**	Chad
CS	Czechoslovakia	**LU**	Luxembourg	**TG**	Togo
CZ	Czech Republic	**LV**	Latvia	**TJ**	Tajikistan
DE	Germany	**MC**	Monaco	**TT**	Trinidad and Tobago
DK	Denmark	**MD**	Republic of Moldova	**UA**	Ukraine
ES	Spain	**MG**	Madagascar	**US**	United States of America
FI	Finland	**ML**	Mali	**UZ**	Uzbekistan
FR	France	**MN**	Mongolia	**VN**	Viet Nam
GA	Gabon				

ARCHITECTURE AND INTERCONNECT SCHEME FOR PROGRAMMABLE LOGIC CIRCUITS

FIELD OF THE INVENTION

The present invention pertains to the field of programmable logic circuits. More particularly, the present invention relates to an architecture and interconnect scheme for programmable logic circuits.

BACKGROUND OF THE INVENTION

When integrated circuits (ICs) were first introduced, they were extremely expensive and were limited in their functionality. Rapid strides in semiconductor technology have vastly reduced the cost while simultaneously increased the performance of IC chips. However, the design, layout, and fabrication process for a dedicated, custom built IC remains quite costly. This is especially true for those instances where only a small quantity of a custom designed IC is to be manufactured. Moreover, the turn-around time (i.e., the time from initial design to a finished product) can frequently be quite lengthy, especially for complex circuit designs. For electronic and computer products, it is critical to be the first to market. Furthermore, for custom ICs, it is rather difficult to effect changes to the initial design. It takes time, effort, and money to make any necessary changes.

In view of the shortcomings associated with custom IC's, field programmable gate arrays (FPGAs) offer an attractive solution in many instances. Basically, FPGAs are standard, high-density, off-the-shelf ICs which can be programmed by the user to a desired configuration. Circuit designers first define the desired logic functions, and the FPGA is programmed to process the input signals accordingly. Thereby, FPGA implementations can be designed, verified, and revised in a quick and efficient manner. Depending on the logic density requirements and production volumes, FPGAs are superior alternatives in terms of cost and time-to-market.

A typical FPGA essentially consists of an outer ring of I/O blocks surrounding an interior matrix of configurable logic blocks. The I/O blocks residing on the periphery of an FPGA are user programmable, such that each block can be programmed independently to be an input or an output and can also be tri-statable. Each logic block typically contains programmable combinatorial logic and storage registers. The combinatorial logic is used to perform boolean functions on its input variables. Often, the registers are loaded directly from a logic block input, or they can be loaded from the combinatorial logic.

Interconnect resources occupy the channels between the rows and columns of the matrix of logic blocks and also between the logic blocks and the I/O blocks. These interconnect resources provide the flexibility to control the interconnection between two

designated points on the chip. Usually, a metal network of lines run horizontally and vertically in the rows and columns between the logic blocks. Programmable switches connect the inputs and outputs of the logic blocks and I/O blocks to these metal lines. Crosspoint switches and interchanges at the intersections of rows and columns are used to switch signals from one line to another. Often, long lines are used to run the entire length and/or breadth of the chip.

The functions of the I/O blocks, logic blocks, and their respective interconnections are all programmable. Typically, these functions are controlled by a configuration program stored in an on-chip memory. The configuration program is loaded automatically from an external memory upon power-up, on command, or programmed by a microprocessor as part of system initialization.

The concept of FPGA was summarized in the sixty's by Minnick who described the concept of cell and cellular array as reconfigurable devices in the following documents: Minnick, R.C. and Short, R.A., "Cellular Linear-Input Logic, Final Report," SRI Project 4122, Contract AF 19(628)-498, Stanford Research Institute, Menlo Park, California, AFCRL 64-6, DDC No. AD 433802 (February 1964); Minnick, R.C., "Cobweb Cellular Arrays," Proceedings AFIPS 1965 Fall Joint Computer Conference, Vol. 27, Part 1 pp. 327-341 (1965); Minnick, R.C. et al., "Cellular Logic, Final Report," SRI Project 5087, Contract AF 19(628)-4233,

Stanford Research Institute, Menlo Park, California, AFCRL 66-613, (April 1966); and Minnick, R.C., "A Survey of Microcellular Research," Journal of the Association for Computing Machinery Vol. 14, No. 2, pp. 203-241 (April 1967). In addition to memory based (e.g., RAM-based, fuse-based, or antifuse-based) means of enabling interconnects between devices, Minnick also discussed both direct connections between neighboring cells and use of busing as another routing technique. The article by Spandorfer, L. M., "Synthesis of Logic Function on an Array of Integrated Circuits," Stanford Research Institute, Menlo Park, Calif., Contract AF 19(628)2907, AFCRL 64-6, DDC No. AD 433802 (November 1965), discussed the use of complementary MOS bi-directional passgate as a means of switching between two interconnect lines that can be programmed through memory means and adjacent neighboring cell interconnections. In Wahlstrom, S. E., "Programmable Logic Arrays - Cheaper by the Millions," Electronics, Vol. 40, No. 25, 11, pp. 90-95 (December 1967), a RAM-based, reconfigurable logic array of a two-dimensional array of identical cells with both direct connections between adjacent cells and a network of data buses is described.

Shoup, R. G., "Programmable Cellular Logic Arrays," Ph.D. dissertation, Carnegie-Mellon University, Pittsburgh, PA (March 1970), discussed programmable cellular logic arrays and reiterates many of the same concepts and terminology of Minnick and recapitulates the array of Wahlstrom. In Shoup's thesis, the

concept of neighbor connections extends from the simple 2-input 1-output nearest-neighbor connections to the 8-neighbor 2-way connections. Shoup further described use of bus as part of the interconnection structure to improve the power and flexibility of an array. Buses can be used to route signals over distances too long, or in inconvenient directions, for ordinary neighbor connections. This is particularly useful in passing inputs and outputs from outside the array to interior cells.

U.S. Patent Number 4,020,469 discussed a programmable logic array that can program, test, and repair itself. U.S. Patent Number 4,870,302 introduced a coarse grain architecture without use of neighbor direct interconnections where all the programmed connections are through the use of three different sets of buses in a channeled architecture. The coarse grain cell (called a Configurable Logical block or CLB) contains both RAM-based logic table look up combinational logic and flip flops inside the CLB where a user defined logic must be mapped into the functions available inside the CLB. U.S. Patent Number 4,935,734 introduced a simple logic function cell defined as a NAND, NOR or similar types of simple logic function inside each cell. The interconnection scheme is through direct neighbor and directional bus connections. U.S. Patent Numbers 4,700,187 and 4,918,440 defined a more complex logic function cell where an Exclusive OR and AND functions and a register bit is available and selectable within the cell. The preferred connection scheme is through direct

neighbor connections. Use of bi-direction buses as connections were also included.

Current FPGA technology has a few shortcomings. These problems are embodied by the low level of circuit utilization given the vast number of transistors available on chip provided by the manufacturers. Circuit utilization is influenced by three factors. The first one at the transistor or fine grain cell level is the function and flexibility of the basic logic element that can be readily used by the users. The second one is the ease in which to form meaningful macro logic functions using the first logic elements with minimum waste of circuit area. The last factor is the interconnections of those macro logic functions to implement chip level design efficiently. The fine grained cell architectures such as those described above, provided easily usable and flexible logical functions for designers at the base logic element level.

However, for dense and complex macro functions and chip level routing, the interconnection resources required to connect a large number of signals from output of a cell to the input(s) of other cells can be quickly exhausted, and adding these resources can be very expensive in terms of silicon area. As a consequence, in fine grained architecture design, most of the cells are either left unused due to inaccessibility, or the cells are used as interconnect wires instead of logic. This adds greatly to routing delays in addition to low logic utilization, or excessive amount of routing resources are added, greatly increasing the circuit size. The coarse

grain architecture coupled with extensive routing buses allows significant improvements for signals connecting outputs of a CLB to inputs of other CLBs. The utilization at the CLB interconnect level is high. However, the difficulty is the partitioning and mapping of complex logic functions so as to exactly fit into the CLBs. If a part of logic inside the CLB is left unused, then the utilization (effective number of gates per unit area used) inside the CLB can be low.

Another problem with prior art FPGAs is due to the fact that typically a fixed number of inputs and a fixed number of outputs are provided for each logic block. If, by happenstance, all the outputs of a particular logic block is used up, then the rest of that logic block becomes useless.

Therefore, there is a need in prior art FPGAs for a new architecture that will maximize the utilization of an FPGA while minimizing any impact on the die size. The new architecture should provide flexibility in the lowest logic element level in terms of functionality and flexibility of use by users, high density per unit area functionality at the macro level where users can readily form complex logic functions with the base logic elements, and finally high percentage of interconnectability with a hierarchical, uniformly distributed routing network for signals connecting macros and base logic elements at the chip level. Furthermore, the new architecture should provide users with the flexibility of having the number of inputs and outputs for

individual logical block be selectable and programmable, and a scalable architecture to accommodate a range of FPGA sizes.

SUMMARY OF THE INVENTION

The present invention relates to an architecture of logic and connection scheme for programmable logic circuits, such as those for field programmable gate arrays (FPGAs). The programmable logic circuit is comprised of a number of cells which perform digital functions on input signals. Depending on user's specific design, certain cells are programmably interconnected to a particular configuration for realizing the desired logic functions.

In the currently preferred embodiment, four logic cells (four two-input one-output logic gates and one D flip-flop) form a logical cluster (i.e. a 2x2 cell array) and four sets of clusters form a logical block (i.e. a 4x4 cell array). Within each cluster, there is a set of five intraconnection lines, called Intraconnection Matrix (I-Matrix), one associated with the output of each one of the four gates and the D flip-flop that is connectable to the input of the other cells. Within each logical block, the I-Matrix within each cluster can be extended to an adjacent cluster through a passgate to form connections within the logical block (to extend the intraconnection range). Inside each logical block, there is an associated set of access lines called Block Connectors (BCs). The block connectors provide access to and connectability between the various cells of that same logical block. In other words, each input and output of each of the cells of a logical block is capable of being connected to a set of block connectors corresponding to that logical block. With the judicious use of I-Matrix and block connectors

within the same logical block, a set of signals can be internally connected without using any resources outside the logical block. A number of programmable switches are used to control which of the block connectors are to be connected together to a set of inputs and/or outputs of the cells inside the logical block for external access connecting to signals outside the current logical block. In other words, the input and/or output pins inside a logical block that are to be externally connected outside of the current logical block are accessed or connected through block connectors within the current logical block.

In order to route signals between the various logical blocks, a uniformly distributed multiple level architecture (MLA) routing network is used to provide connectability between each of the individual sets of block connectors. Programmable switches are implemented to control which of the first level MLA routing network lines are to be connected together. Additional programmable switches are used to control which of the block connectors are to be connected to specific first level MLA routing lines. For example, the switches can be programmed to allow an originating cell belonging to one logical block to be connected to a destination cell belonging to a different logical block. This can be accomplished by connecting the originating cell through one or more of its block connectors, onto the first level MLA, depending on the distance, other level(s) of MLA, and down through descending levels of MLAs back to the first level MLA, and finally

through the block connector of the destination cell. Thereby, the block connectors and first level of MLA routing network provide interconnectability for an 8x8 cell array, called a block cluster.

In the present invention, larger cell arrays can be interconnected by implementing additional levels of MLA routing networks. For example, connectability for a 16x16 cell array, called a block sector, can be achieved by implementing a second level of MLA routing network lines to provide connectability between the various first level of MLA routing lines thereby making connections between different block clusters. Each level of MLA has a corresponding number of switches for providing programmable interconnections of the routing network of that level. Additional switching exchange networks are used to provide connectability between the various levels of MLAs.

In one embodiment, switches are used to provide connectability between two different sets of block connectors. Moreover, switches can be included to provide connectability between different sets of MLA routing lines of a particular level of MLAs. This provides for increased routing flexibility.

In the present invention, all MLA routing network lines are bi-directional. The switches are comprised of programmable bi-directional passgates. For increased number of levels, drivers may be necessary for providing the necessary switching speed for driving the routing lines, passgates, and associated loads, etc. In one embodiment, switches are used to provide programmable

connectability amongst various sets of block connectors. Additional switches can be implemented to provide programmable connectability amongst various sets of the first level of MLA. This scheme can be repeated for higher levels of MLAs.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention is illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings and in which like reference numerals refer to similar elements and in which:

Figure 1 is a block diagram of a field programmable gate array logic upon which the present invention may be practiced.

Figure 2A shows one example of an individual cell.

Figure 2B shows another example of an individual cell.

Figure 3A shows a logical cluster.

Figure 3B shows the extension of I-matrix intraconnections of a logical cluster to a neighboring logical cluster.

Figure 4A shows an example of a logical cluster with vertical block connectors.

Figure 4B shows an example of a logical cluster with horizontal block connectors.

Figure 5A shows the eight block connector to level 1 MLA exchange networks associated with a logical block and level 1 MLA turn points.

Figure 5B shows a level 1 MLA turn point.

Figure 5C shows an exchange network.

Figure 6 shows the routing network for a block cluster.

Figure 7A shows the block diagram of a block sector.

Figure 7B shows a level 1 to level 2 MLA routing exchange network.

Figure 8A shows a sector cluster.

Figure 8B shows a level 2 to level 3 MLA routing exchange network.

DETAILED DESCRIPTION

An architecture and interconnect scheme for programmable logic circuits is described. In the following description, for purposes of explanation, numerous specific details are set forth, such as combinational logic, cell configuration, numbers of cells, etc., in order to provide a thorough understanding of the present invention. It will be obvious, however, to one skilled in the art that the present invention may be practiced without these specific details. In other instances, well-known structures and devices are shown in block diagram form in order to avoid unnecessarily obscuring the present invention. It should also be noted that the present invention pertains to a variety of processes including but not limited to static random access memory (SRAM), dynamic random access memory (DRAM), fuse, anti-fuse, erasable programmable read only memory (EPROM), electrically erasable programmable read only memory (EEPROM), FLASH, and ferroelectric processes.

Referring to Figure 1, a block diagram of a field programmable gate array logic upon which the present invention may be practiced is shown as 100. The I/O logical blocks 102, 103, 111 and 112 provide an interface between external package pins of the FPGA and the internal user logic either directly or through the I/O to Core interface 104, 105, 113, and 114. Four interface blocks 104, 105, 113, and 114 provide decoupling between core 106 and the I/O logic 102, 103, 111, and 112. Core 106 is

comprised of a number of clusters 107 which are intraconnected by I-Matrix 101 and interconnected by MLA routing network 108.

Control/programming logic 109 is used to control all of the bits for programming the bit and word lines. For anti-fuse or fuse technology, high voltage/current is applied to either zap or connect a fuse. For EEPROM, Flash, or ferroelectric technology, there is an erase cycle followed by a programming cycle for programming the logic states of the memory bits. In order to minimize skewing, a separate clock/reset logic 110 is used to provide clock and reset lines on a group basis.

In the currently preferred embodiment, each of the clusters 107 is comprised of a 2x2 hierarchy of four cells, called a logical cluster. Figures 2A and 2B show examples of individual cells 200 and 250. Cell 200 performs multiple logic functions on two input signals (A and B) and provides an output signal X. In the currently preferred embodiment, cell 200 is comprised of an XOR gate 201, a two-input NAND gate 202, and a two-input NOR gate 203. It should be noted, however, that in other embodiments, cell 200 can include various other types and/or combinations of gates. Cell 250 is comprised of cell 200 coupled with a D flip flop cell 260. The output X of cell 200 can be programmed to connect directly to the data input D of the D flip flop gate 204 by activating switch 218. The data input D can be accessed as a third input of the combined cell 250.

Each of the two input signals A and B and the D input of D flip-flop can be inverted or non-inverted, depending on the states of switches 206-211. Activating switches 206, 208 and 210 causes signals A, B and D to be driven by drivers 212-214 to gates 201-204 in a non-inverted fashion. Activating switches 207, 209, and 211 causes the input signals A, B and D to be inverted by inverters 215-217 before being passed to gates 201-204. The six switches 212-217 can individually be turned on and off as programmed by the user.

Note that the XOR gate 201, NAND gate 202, and NOR gate 203 can also be used to perform XNOR, AND and OR by propagating the output signal to the next stage, whereby the signal can be inverted as discussed above.

Three switches 219-221 are respectively coupled to the outputs of the three gates 201-203. Again, these switches are programmable by the user. Thereby, the user can specify which of the outputs from the gates 201-203 is to be sent to driver 224 as the output X from cell 200.

The aforementioned switches 206-211, 218-221 are comprised of bi-directional, program-controlled passgates. Depending on the state of the control signal, the switches are either conducting (i.e. passes a signal on the line) or non-conducting (i.e. does not pass the signal on the line). Switches mentioned in the following sections are similarly comprised of program-controlled passgates.

Referring now to Figure 3A, a logical cluster 107 is shown. In the currently preferred embodiment, logical cluster 107 is comprised of four cells 301-304 and a D flip-flop 305, twenty five switches 306-330, and five intraconnection lines 331-335. The Intraconnection lines 331-335 and switches 306-330 form the I-Matrix. I-Matrix provide connectability of the output, X, of each of the four cells 301-304, and the output X of the D flip-flop 305 to at least one input of each of the other three cells and the D flip-flop. For example, the output X of cell 301 can be connected to input A of cell 302 by enabling switches 306 and 307. Likewise, the output X of cell 301 can be connected to input B of cell 303 by enabling switches 306 and 310. Output X of cell 301 can be connected to input A of cell 304 by enabling switches 306 and 308. Output X of cell 301 can be connected to input D of the D flip-flop cell 305 by enabling switches 306 and 309.

Similarly, the output X from cell 302 can be connected to input A of cell 301 by enabling switches 311 and 312. The output X from cell 302 can be connected to input A of cell 303 by enabling switches 311 and 315. The output X from cell 302 can be connected to input B of cell 304 by enabling switches 311 and 313. Output X of cell 302 can be connected to input D of the D flip-flop cell 305 by enabling switches 311 and 314.

Similarly, the output X from cell 303 can be connected to input B of cell 301 by enabling switches 326 and 327. The output X from cell 303 can be connected to input A of cell 302 by

enabling switches 326 and 328. The output X from cell 303 can be connected to input B of cell 304 by enabling switches 326 and 329. Output X of cell 303 can be connected to input D of the D flip-flop cell 305 by enabling switches 326 and 330.

For cell 304, the output X from cell 304 can be connected to input B of cell 301 by enabling switches 316 and 317. The output X from cell 304 can be connected to input B of cell 302 by enabling switches 316 and 318. The output X from cell 304 can be connected to input A of cell 303 by enabling switches 316 and 319. Output X of cell 304 can be programmably connected to input D of the D flip-flop cell 305 by enabling switch 218 in Figure 2A.

With respect to cell 305, its output is connectable to the A input of cell 301 by enabling switches 320 and 321; the B input of cell 302 by enabling switches 320 and 322; the B input of cell 303 by enabling switches 320 and 325; the A input of cell 304 by enabling switches 320 and 323; and the D input of cell 305 itself by enabling switches 320 and 324.

It can be seen that each output of the cells 301-304 and of the D flip-flop 305 is connectable to the input of each of its neighboring cells and/or flip-flop inside the cluster.

In the currently preferred embodiment of the present invention, each logical cluster is connectable to all the other logical clusters inside each logical block through passgate switches extending the I-Matrix from neighboring clusters inside each

logical block. Figure 3B illustrates the extension of I-Matrix intraconnection lines 331-335 of the cells 301-304 and the D flip-flop 305 of a logical cluster 107 to a neighboring logical cluster 107 through the passgate switches 336-355 within the same logical block.

In the currently preferred embodiment of the present invention, each logical block is connectable to all the other logical blocks of the FPGA. This is accomplished by implementing an architecture with multiple layers of interconnections. It is important to note that this multiple layers routing architecture is a conceptual hierarchy, not a process or technology hierarchy and is hence readily implementable with today's silicon process technology. The bottom most layer of interconnections is referred to as the "block connectors". A set of block connectors provides the access and interconnections of signals within an associated logical block (which is consisted of four logical clusters or 16 cells). Thereby, different sets of logical clusters within the same logical block are connectable to any of the other logical clusters within that group through the use of extended I-Matrix and/or block connectors. Again, programmable bi-directional passgates are used as switches to provide routing flexibility to the user.

The next level of connections is referred to as the "level 1 Multiple Level Architecture (MLA)" routing network. The level 1 MLA routing network provides the interconnections between several sets of block connectors. Programmable passgates

switches are used to provide users with the capability of selecting which of the block connectors are to be connected. Consequently, a first logical block from one set of logical block groups is connectable to a second logical block belonging to the same group. The appropriate switches are enabled to connect the block connectors of the first logical block to the routing lines of the level 1 MLA routing network. The appropriate switches of the level 1 MLA routing network are enabled to provide the connections to the block connectors of the second logical block to the routing lines of the level 1 MLA routing network. The appropriate switches are enabled to connect the routing lines of the level 1 MLA routing network that connected to the block connectors of the first and the second logical blocks. Furthermore, the user has the additional flexibility of programming the various switches within any given logical block to effect the desired intraconnections between each of the cells of any logical block.

The next level of connections is referred to as the "level 2 Multiple Level Architecture (MLA)" routing network. The level 2 MLA provides the interconnections to the various level 1 MLA to effect access and connections of a block cluster. Again, bi-directional passgate switches are programmed by the user to effect the desired connections. By implementing level 2 MLA routing network, programmable interconnections between even larger numbers of logical blocks is achieved.

Additional levels of MLA routing networks can be implemented to provide programmable interconnections for ever increasing numbers and groups of logical blocks, block clusters, block sectors, etc. Basically, the present invention takes a three dimensional approach for implementing routing. Signals are routed amongst the intraconnections of a logical block. These signals can then be accessed through block connectors and routed according to the programmed connections of the block connectors. If needed, signals are "elevated" to the level 1 MLA, routed through the level 1 MLA routing network, "de-elevated" to the appropriate block connectors, and then passed to the destination logical block.

If level 2 MLA routing network is required, some of the signals are elevated a second time from a level 1 MLA routing network line to the level 2 MLA routing network, routed to a different set of level 2 MLA routing network line, and "de-elevated" from the level 2 MLA routing network line to a Level 1 MLA routing network line. Thereupon, the signals are "de-elevated" a second time to pass the signal from the level 1 MLA to the appropriate block connectors of the destination logical block. This same approach is performed for level 3, 4, 5, etc. MLAs on an as needed basis, depending on the size and density of the FPGA. Partial level n MLA can be implemented using the above discussed method to implement a FPGA with a given cell array count.

Figure 4A shows an example of a logical cluster and the associated vertical block connectors within the logical block. In the currently preferred embodiment, each cell in a logical cluster is accessible from the input by two vertical block connectors and each output of the cell in a logical cluster is accessible to two of the vertical block connectors. For example, input A of cell 301 is accessible to the vertical block connectors 451 (BC-V11) and 453 (BC-V21) through switches 467, 462 respectively, input B of cell 301 is accessible to the vertical block connectors 452 (BC-V12) and 454 (BC-V22) through switches 466, 468 respectively, output X of cell 301 is accessible to the vertical block connectors 455 (BC-V31) and 458 (BC-V42) through switches 460, 459 respectively. Input A of cell 302 is accessible to the vertical block connectors 453 (BC-V21) and 455 (BC-V31) through switches 463, 464 respectively, input B of cell 302 is accessible to the vertical block connectors 454 (BC-V22) and 456 (BC-V32) through switches 469, 470 respectively, output X of cell 302 is accessible to the vertical block connectors 452 (BC-V12) and 457 (BC-V41) through switches 461, 465 respectively. Input A of cell 303 is accessible to the vertical block connectors 451 (BC-V11) and 453 (BC-V21) through switches 485, 476 respectively, input B of cell 303 is accessible to the vertical block connectors 452 (BC-V12) and 454 (BC-V22) through switches 480, 476 respectively, output X of cell 303 is accessible to the vertical block connectors 455 (BC-V31) and 458 (BC-V42) through switches 472, 471 respectively. The

input A of cell 304 is accessible to the vertical block connectors 453 (BC-V21) and 455 (BC-V31) through switches 477, 478 respectively, input B of cell 304 is accessible to the vertical block connectors 454 (BC-V22) and 456 (BC-V32) through switches 482, 484 respectively, output X of cell 304 is accessible to the vertical block connectors 452 (BC-V12) and 457 (BC-V41) through switches 475, 474 respectively. D flip-flop cell 305 input is accessible to the vertical block connectors 454 (BC-V22) and 455 (BC-V31) through switches 473, 479 respectively, output X of cell 305 is accessible to the vertical block connectors 452 (BC-V12) and 457 (BC-V41) through switches 483, 486 respectively.

In similar fashion, Figure 4B shows the possible connections corresponding to horizontal block connectors and the logical cluster shown in Figure 4A. Input A of cell 301 is accessible to the horizontal block connectors 402 (BC-H12) and 404 (BC-H22) through switches 409, 413 respectively, input B of cell 301 is accessible to the horizontal block connectors 401 (BC-H11) and 403 (BC-H21) through switches 415, 416 respectively, output X of cell 301 is accessible to the horizontal block connectors 405 (BC-H31) and 408 (BC-H42) through switches 421, 428 respectively. Input A of cell 302 is accessible to the horizontal block connectors 402 (BC-H12) and 404 (BC-H22) through switches 411, 414 respectively, input B of cell 302 is accessible to the horizontal block connectors 401 (BC-H11) and 403 (BC-H21) through switches 433, 417 respectively, output X of cell 302 is accessible

to the horizontal block connectors 405 (BC-H31) and 408 (BC-H42) through switches 418, 424 respectively. Input A of cell 303 is accessible to the horizontal block connectors 404 (BC-H22) and 406 (BC-H32) through switches 419, 426 respectively, input B of cell 303 is accessible to the horizontal block connectors 403 (BC-H21) and 405 (BC-H31) through switches 420, 425 respectively, output X of cell 303 is accessible to the horizontal block connectors 402 (BC-H12) and 407 (BC-H41) through switches 410 427 respectively. The input A of cell 304 is accessible to the horizontal block connectors 404 (BC-H22) and 406 (BC-H32) through switches 422,430 respectively, input B of cell 304 is accessible to the horizontal block connectors 403 (BC-H21) and 405 (BC-H31) through switches 423, 429 respectively, output X of cell 304 is accessible to the horizontal block connectors 402 (BC-H12) and 407 (BC-H41) through switches 412, 434 respectively. D flip-flop cell 305 input is accessible to the horizontal block connectors 403 (BC-H21) and 406 (BC-H32) through switches 436, 431 respectively, output X of cell 305 is accessible to the horizontal block connectors 401 (BC-H11) and 408 (BC-H42) through switches 432, 435 respectively.

Figures 4A and 4B illustrate the vertical and horizontal block connectors accessing method to the upper left (NW) logical cluster inside a logical block in the currently preferred embodiment. The lower left (SW) cluster has the identical accessing method to the vertical block connectors as those of the

NW cluster. The upper right (NE) cluster has similar accessing method to those of the NW cluster with respect to the vertical block connectors except the sequence of vertical block connector access is shifted. The vertical block connectors 451-458 can be viewed as chained together as a cylinder (451, 452, ..., 458). Any shift, say by 4, forms a new sequence: (455, 456, 457, 458, 451, 452, 453, 454). Instead of starting with vertical block connectors 451 and 453 accessing by cell 301 in the NW cluster as illustrated in Figures 4A, the cell 301 in the NE cluster is accessible to VBCs 455 and 457. The numbering is "shifted" by four. The access labeling of the lower right (SE) cluster to the VBCs is identical to those of NE cluster.

Similarly, the horizontal block connectors access to the NW cluster is identical to those of the NE cluster and the SW cluster is identical to the SE cluster while the horizontal block connectors access to the SW cluster is shifted by four compared with those of NW cluster.

In the currently preferred embodiment, sixteen block connectors are used per logical block (i.e. four clusters, or a 4x4 cell array). Adding a level 1 MLA routing network allows for the connectability for a block cluster (an 8x8 cell array). Adding level 2 MLA routing network increases the connectability to a block sector (16x16 cell array). Additional levels of MLA routing network increases the number of block sectors by factors of four while the length (or reach) of each line in the MLA routing

network increases by factors of two. The number of routing lines in the level 2 MLA is increased by a factor of two; since the number of block sectors increased by a factor of four, on a per unit area basis, the number of routing lines in the next level of hierarchy actually decreases by a factor of two.

Figure 5A shows a logical block with associated sixteen block connectors and level 1 MLA routing lines associated with the logical block. The sixteen block connectors 501-516 are depicted by heavy lines whereas the sixteen level 1 MLA routing network lines 517-532 are depicted by lighter lines. Note that the length or span of the block connectors terminates within the logical block while the length of the level 1 MLA routing network lines extends to neighboring logical blocks (twice the length of the block connectors).

Both block connectors and level 1 MLA routing network lines are subdivided into horizontal and vertical groups: vertical block connectors 501-508, horizontal block connectors 509-516, vertical level 1 MLA routing network lines 517-524, and horizontal level 1 MLA routing network lines 525-532.

In the currently preferred embodiment, there are twenty four level 1 MLA turn points for the sixteen level 1 MLA routing network lines within the logical block. In Figure 5A, the twenty four turn points are depicted as clear dots 541-564.

A MLA turn point is a programmable bi-directional passgate for providing connectability between a horizontal MLA routing

network line and a vertical MLA routing network line. For example, enabling level 1 MLA turn point 541 causes the horizontal level 1 MLA routing network line 526 and vertical level 1 MLA routing network line 520 to become connected together. Figure 5B shows level 1 MLA turn point 541. Switch 583 controls whether level 1 MLA routing network line 526 is to be connected to level 1 MLA routing network line 520. If switch is enabled, then level 1 MLA routing network line 526 is connected to level 1 MLA routing network line 520. Otherwise, line 526 is not connected to line 520. Switch 583 is programmable by the user. The turn points are placed as pair-wise groups with the objective of providing switching access connecting two or more block connectors first through the block connector to level 1 MLA exchange networks and then connecting selected level 1 MLA routing lines by enabling the switches. The level 1 MLA lines are used to connect those block connectors that reside in separate logical blocks within the same block cluster.

Referring back to Figure 5A, there are eight block connector to level 1 MLA exchange networks 533-540 for each logical block. These exchange networks operate to connect certain block connectors to level 1 MLA lines as programmed by the user. Figure 5C shows the exchange network 537 in greater detail. The block connector to level 1 MLA routing exchange network has eight drivers 575-582. These eight drivers 575-582 are used to provide bi-directional drive for the block connectors 501, 502 and

level 1 MLA lines 517, 518. For example, enabling switch 565 causes the signal on block connector 501 to be driven by driver 575 onto the level 1 MLA line 517. Enabling switch 566 causes the signal on level 1 MLA line 517 to be driven by driver 576 onto the block connector 501. Enabling switch 567 causes the signal on block connector 501 to be driven by driver 577 onto the level 1 MLA line 518. Enabling switch 568 causes the signal on level 1 MLA line 518 to be driven by driver 578 onto the block connector 501.

Similarly, enabling switch 569 causes the signal on block connector 502 to be driven by driver 579 onto the level 1 MLA line 517. Enabling switch 570 causes the signal on level 1 MLA line 517 to be driven by driver 580 onto the block connector 502. Enabling switch 571 causes the signal on block connector 502 to be driven by driver 581 onto the level 1 MLA line 518. Enabling switch 572 causes the signal on level 1 MLA line 518 to be driven by driver 582 onto the block connector 502. Switch 573 is used to control whether a signal should pass form one block connector 501 to the adjacent block connector 584 belonging to the adjacent logical block.

Likewise, switch 574 is used to control whether a signal should pass form one block connector 502 to the adjacent block connector 585 belonging to the adjacent logical block.

Figure 6 shows the routing network for a block cluster. The block cluster is basically comprised of four logical blocks which

can be interconnected by the level 1 MLA exchange networks 533-540. It can be seen that there are thirty-two level 1 MLA routing network lines.

Figure 7A shows the block diagram for a block sector. The block sector is comprised of four block clusters 701-704. As discussed above, the block clusters are interconnected by block connectors and level 1 MLA routing network lines. In addition, the block sector is also comprised of sixty-four level 2 MLA routing network lines and sixty-four level 2 to level 1 exchange networks to provide connectability between level 1 MLA routing network and level 2 MLA routing network. The level 1 to level 2 MLA routing exchange networks are depicted by rectangles in Figure 7A. Furthermore, there are forty-eight level 2 MLA turn points associated with each of the four logical blocks within the block sector. Consequently, there are one hundred and ninety-two level 2 MLA turn points for the block sector.

Figure 7B shows a sample level 1 to level 2 MLA routing exchange network 705. It can be seen that switch 710 is used to control whether a signal should pass between level 1 MLA line 709 and level 2 MLA line 708. Switch 711 is used to control whether a signal should pass between level 1 MLA line 709 and level 2 MLA line 707. Switch 712 is used to control whether a signal should pass between level 1 MLA line 706 and level 2 MLA line 708. Switch 713 is used to control whether a signal should pass between level 1 MLA line 706 and level 2 MLA line 707.

Switch 714 is used to control whether a signal should pass form one level 1 MLA line 709 to the adjacent level 1 MLA line 716 belonging to the adjacent block cluster. Likewise, switch 715 is used to control whether a signal should pass form one level 1 MLA line 706 to the adjacent level 1 MLA line 715 belonging to the adjacent block cluster.

Figure 8A shows a sector cluster. The sector cluster is comprised of four block sectors 801-804 with their associated block connectors, level 1, and level 2 MLA routing network lines and exchange networks. In addition, there are one hundred and twenty-eight level 3 MLA routing network lines, providing connectability between the level 2 MLA lines that belong to different block sectors 801-804 within the same sector cluster 800. There are ninety-six level 3 MLA turn points associated with the level 3 MLA lines for each of the block sector 801-804 (i.e. three hundred and eighty-four total level 3 MLA turn points for the sector cluster). Furthermore, there are thirty-two level 2 to level 3 MLA routing exchange networks associated with each of the four block sector 801-804. Hence, there are total of one hundred and twenty-eight level 3 MLA routing exchange network for providing programmable connectability between the various level 2 and level 3 MLA lines.

Figure 8B shows an example of a level 2 to level 3 MLA routing exchange network 805. It can be seen that enabling switch 810 causes a signal on the level 2 MLA line 808 to be

connected to the level 3 MLA line 806. Disabling switch 810 disconnects the level 2 MLA line 808 from the level 3 MLA line 806. Enabling switch 811 causes a signal on the level 2 MLA line 808 to be connected to the level 3 MLA line 807. Disabling switch 811 disconnects the level 2 MLA line 808 from the level 3 MLA line 807. Likewise, enabling switch 812 causes a signal on the level 2 MLA line 809 to be connected to the level 3 MLA line 806. Disabling switch 812 disconnects the level 2 MLA line 809 from the level 3 MLA line 806. Enabling switch 813 causes a signal on the level 2 MLA line 809 to be connected to the level 3 MLA line 807. Disabling switch 813 disconnects the level 2 MLA line 809 from the level 3 MLA line 807.

In the present invention, larger and more powerful FPGAs can be achieved by adding additional logic sector clusters which are connected by additional levels of MLA routing networks with the corresponding MLA turn points and exchange networks.

In one embodiment of the present invention, each of the five I-Matrix lines (331-335, Figure 3A) can be extended to provide connectability between two adjacent I-Matrix lines belonging to two different clusters. The passgate switches 336-340, 341-345, 346-350, and 351-355 in Figure 3B are examples of four different sets of I-Matrix line extension switches. This provides further flexibility by providing the capability of routing a signal between two adjacent clusters without having to be routed through the use of block connectors.

Similarly, block connectors can be extended to provide connectability between two adjacent block connectors belonging to two different logical blocks. Switch 573 of Figure 5C illustrates such block connector extension connecting block connector 501 to block connector 584 through switch 573. This provides further flexibility by providing the capability of routing a signal between two adjacent logical blocks without having to be routed through the level 1 MLA lines and associated MLA exchange networks. This concept can be similarly applied to the level 1 MLA lines as well. Switch 714 of Figure 7B shows an example where level 1 MLA line 709 is extended to connect to level 1 MLA line 716 by enabling switch 714. This provides further flexibility by providing the capability of routing a signal between two adjacent block clusters without having to be routed through the level 2 MLA lines and associated MLA exchange networks.

Thus, an architecture with an intraconnect and interconnect scheme for programmable logic circuits is disclosed.

CLAIMS

What is claimed is:

1. A programmable logic circuit comprising:

an input means for inputting a signal to said programmable logic circuit;

a plurality of cells coupled to said input means, each of said cells capable of performing a digital function to said signal;

a first set of routing lines for coupling a plurality of cells to form a logical cluster of cells, wherein each output of each of said cells of said logical cluster can be programmably coupled to at least one input of each of the other cells of said logical cluster;

a second set of routing lines for coupling a plurality of logical clusters to form a logical block of logical clusters of cells, wherein each input and each output of every cell of said logical block can be programmably coupled to said second set of routing lines.

2. The programmable logic circuit of Claim 1, wherein at least one output of said cells of said logical cluster can be programmably coupled to at least one input of said cells of said logical cluster.

3. The programmable logic circuit of Claim 1, wherein said second set of routing lines can be programmably selected to conduct either output signals or input signals.

4. The programmable logic circuit of Claim 1 further comprising:

a first set of switches for coupling an output of one cell of a first logical cluster to an input of at least one cell of said first logical cluster by selecting one routing line of said first set of routing lines associated with said first logical cluster;

a second set of switches for coupling outputs or inputs of said cells of said first logical block to said routing lines of said associated second set of routing lines, wherein said second set of switches can be used to couple one routing line of said second set of routing lines to another routing line of said second set of routing lines;

a third set of switches for coupling one routing line of said first set of routing lines of said associated first logical cluster to an adjacent routing line of said first set of routing lines of an adjacent second logical cluster wherein said first logical cluster and said second logical cluster are of a same logical block;

a fourth set of switches for coupling one routing line of said second set of routing lines of said associated first logical block to an adjacent routing line of said second set of routing lines of said adjacent second logical block.

5. The programmable logic circuit of Claim 4 further comprising:

a plurality of inverting and non-inverting drivers coupled to each of said cells for inverting input signals to said cells;

a plurality of programmable switches for determining whether said input signals to said cells are to be routed through said drivers.

6. The programmable logic circuit of Claim 5, wherein said first set of switches, said second set of switches, said third set of switches, said fourth set of switches, and said plurality of programmable switches are comprised of programmable passgates.

7. The programmable logic circuit of Claim 6, wherein said cells of said logical cluster includes four two-input combination logic and a flip-flop.

8. The programmable logic circuit of Claim 7, wherein said logical block includes four of said logical clusters, each of said logical clusters is comprised of at least four of said cells.

9. The programmable logic circuit of Claim 8, wherein additional sets of logical blocks are interconnected by additional sets of hierarchical routing networks.

10. The programmable logic circuit of Claim 9, wherein said first set of routing lines and said second set of routing lines are comprised of one-bit wide bi-directional data buses.

11. The programmable logic circuit of Claim 10 further comprising a plurality of drivers for bi-directionally driving said data buses.

12. The programmable logic circuit of Claim 11, wherein said circuit is implemented by at least one of SRAM, DRAM, fuse, anti-fuse, ferroelectric, EEPROM, EPROM, and FLASH processes.

13. A field programmable gate array comprising:

an input/output interface for inputting signals to said field programmable gate array and for outputting signals from said field programmable gate array;

a plurality of logical clusters of cells coupled to said input/output interface, said cells performing digital processing on signals input to said field programmable gate array;

a plurality of block connectors for coupling inputs and outputs of cells of one of said logical clusters to inputs and outputs of cells of another of said logical clusters, wherein at least two of said logical clusters are associated with a same logical block;

a first set of switches for programmably coupling outputs or inputs of said cells of said logical block to said block connectors;

a plurality of a first level of routing network lines for coupling inputs and outputs of cells of one of said logical blocks to inputs and outputs of cells of another of said logical blocks;

a second set of switches for programmably coupling routing lines of said first level routing network lines;

a plurality of a second level of routing network lines for coupling a first routing network line of one of said plurality of first level routing network lines to a second routing network line of one of said plurality of first level routing network lines;

a third set of switches for programmably coupling routing lines of said second level routing network lines;

a plurality of a third level of routing network lines for connecting a first routing network line of one of said plurality of second level routing network lines to a second routing network line of one of said plurality of second level routing network lines;

a fourth set of switches for programmably coupling routing lines of said third level routing network lines;

a fifth set of switches for programmably coupling at least one routing line of said first level routing network lines to one routing line of said second level routing network lines;

a sixth set of switches for programmably coupling at least one routing line of said second level routing network lines to one routing line of said third level routing network lines;

14. The field programmable gate array of Claim 13 further comprising:

a plurality of intraconnections for coupling an output of one of said cells to at least one input of other cells of a same logical cluster;

a seventh set of switches for programmably controlling said intraconnections of said logical clusters.

15. The field programmable gate array of Claim 14 further comprising:

a plurality of inverting and non-inverting drivers coupled to inputs of said cells;

an eighth set of switches for programmably routing signals through said drivers before being input to said cells.

16. The field programmable gate array of Claim 15, wherein said first, said second, said third, said fourth, said fifth, said sixth, said seventh, and said eighth set of switches are comprised of bi-directional passgates.

17. The field programmable gate array of Claim 16, wherein said first, said second, and said third level of routing networks are comprised of one-bit wide, bi-directional data buses.

18. The field programmable gate array of Claim 17 further comprising a plurality of drivers for driving said data buses, wherein said drivers can drive said data buses bi-directionally.

19. The field programmable gate array of Claim 18, wherein said gate array is implemented by at least one of SRAM, DRAM, fuse, anti-fuse, ferroelectric, EEPROM, EPROM, and FLASH processes.

20. A field programmable gate array comprising:

an I/O providing a signal interface for said field programmable gate array;

a plurality of cells coupled to said I/O, said cells digitally processing input signals to said field programmable gate array;

a plurality of sets of block connectors for coupling said plurality of cells, wherein each set of block connectors provides coupling for a logical block comprising a 4x4 cell array;

a first set of switches for programmably coupling block connectors corresponding to each set of block connectors associated with said logical block;

a plurality of sets of a first level routing network for coupling said plurality of sets of block connectors, wherein each set of said first level routing network lines provides coupling for a block cluster comprising an 8x8 cell array;

a second set of switches for programmably coupling routing network lines corresponding to each set of said first level routing network;

a second level routing network for coupling said plurality of sets of said first level routing network, wherein said second level routing network provides coupling for a block sector comprising a 16x16 cell array;

a third set of switches for programmably coupling routing network lines corresponding to each set of said second level routing network;

a fourth set of switches for programmably coupling block connectors corresponding said set of block connectors to routing network lines corresponding to said first level routing network;

a fifth set of switches for programmably coupling routing network lines corresponding the said first level routing network to routing network lines corresponding to said second level routing network;

21. The field programmable gate array of Claim 20 further comprising:

a plurality of intraconnection matrix lines providing coupling for a logical cluster comprising a 2x2 cell array;

a sixth set of switches for programmably controlling said intraconnection matrix lines, wherein each output of each cell of said logical cluster can be programmably connected to at least one

input of each of the other cells of said logical cluster and at least one input of an output cell.

22. The field programmable gate array of Claim 21 further comprising:

a plurality of drivers for driving signals prior to input to said cells wherein said drivers can be either inverting or non-inverting;

a seventh set of switches for programmably controlling which of said signals are to be routed through said drivers.

23. The field programmable gate array of Claim 22, further comprising an eighth set of switches for programmably coupling block connectors corresponding to a first set of block connectors to block connectors corresponding to a second set of block connectors.

24. The field programmable gate array of Claim 23 further comprising a ninth set of switches for programmably coupling routing network lines corresponding to a first set of said first level routing network to routing network lines corresponding to a second set of said first level routing network.

25. The field programmable gate array of Claim 24 further comprising additional levels of routing networks, wherein said cell

arrays are increased by a factor of four for each additional level of routing networks and the number of routing network lines are increased by a factor of two for each additional level of routing networks.

26. The field programmable gate array of Claim 25, wherein said additional levels of routing network lines and said cell arrays can be sized with partial levels of routing network lines.

27. The field programmable gate array of Claim 25, wherein said block connectors, said first level of routing network lines, and said second level of routing network lines are bi-directional.

28. The field programmable gate array of Claim 27, wherein said first, said second, said third, said fourth, said fifth, said sixth, said seventh, said eighth, and said ninth sets of switches are comprised of programmable, bi-directional passgates.

29. A method of processing signals in a field programmable gate array, comprising the steps of:

inputting a signal to said field programmable gate array;

coupling a plurality of cells together, said cells digitally processing signals input to said field programmable gate array;

providing a plurality of intraconnection matrix lines for coupling a logical cluster comprising a set of said plurality of cells;

programmably controlling said intraconnection matrix lines, wherein each output of each cell of said logical cluster can be programmably coupled to at least one input of each of the other cells of said logical cluster;

providing coupling for a logical block comprising a set of said logical clusters by a plurality of block connectors;

programmably connecting block connectors corresponding to each set of block connectors associated with said logical block;

coupling said plurality of sets of block connectors by a plurality of sets of a first level routing network lines, wherein each set of said first level routing network lines provides coupling for a block cluster;

programmably connecting routing network lines corresponding to each set of said first level routing network lines;

providing a second level routing network lines for coupling said plurality of sets of said first level routing network lines, wherein said second level routing network lines provides coupling for a block sector comprising a cell array comprising a set of said block clusters;

programmably connecting routing network lines corresponding to each set of said second level routing network lines;

programmably connecting said block connectors corresponding the said set of block connectors to routing network lines corresponding to said first level routing network lines;

programmably connecting routing network lines corresponding the said first level routing network lines to routing network lines corresponding to said second level routing network lines;

programmably connecting block connectors corresponding to a first set of block connectors to block connectors corresponding to a second set of block connectors;

programmably connecting routing network lines corresponding to a first set of said first level routing network lines to routing network lines corresponding to a second set of said first level routing network lines.

30. The method of Claim 27 further comprising the steps of:

implementing additional levels of routing network lines, wherein said cell arrays are increased by a factor proportion to an area increase of said cell arrays for each additional levels of routing network lines, the number of said routing network lines are increased by a factor proportional with the length increase of the said cell arrays for each additional levels of routing network lines;

implementing additional set of switches for programmably connecting routing network lines corresponding to each set of said additional level routing network lines and for programmably connecting routing network lines corresponding to a first set of said additional level routing network lines to routing network lines corresponding to a second set of said level routing network lines that is one level below the hierarchy of said additional level routing network lines;

31. The method of Claim 30, wherein said additional levels of routing network lines and said cell arrays can be sized with partial levels of routing network lines, depending upon the cell array size.

32. The method of Claim 30, wherein said block connectors, said first level of routing network lines, said second level of routing network lines, and additional levels of routing network lines, are bi-directional;

33. The method of Claim 30, wherein said first, said second, said third, said fourth, said fifth, said sixth, said seventh, said eighth, and said additional sets of switches are comprised of programmable, bi-directional passgates.

34. The method in accordance with Claim 30 further comprising the step of adding levels of routing networks, wherein new cell arrays are increased by a factor proportional to an area increase of said cell arrays and for each additional level of routing network lines, the number of said routing network lines are increased by a factor proportional with the length increase of said cell arrays.

35. In a programmable logic circuit having a plurality of cells for digitally processing electrical signals, a method of connecting said plurality of cells to process signals input to said programmable logic circuit according to logic functions defined by a user, said method comprising the steps of:

forming a logical cluster by providing a first set of intraconnection lines for coupling a set of said cells, wherein each output of each of said cells of said logical cluster can be programmably coupled to at least one input of each of the other cells of said logical cluster;

forming a logical block by providing a second set of block connectors for coupling a set of logical clusters, wherein each input and each output of every cell of said logical block can be programmably coupled to said second set of block connectors;

selecting one of said intraconnection lines of said first set of intraconnection lines corresponding to a first logical cluster;

controlling a first switch to couple an output of one cell of said first logical cluster to an input of another cell of said first logical cluster;

controlling a second switch to coupled an input or output of one of said cells of said first logical block to a block connector of said second set of block connectors;

controlling third switch for coupling one intraconnection line of said first set of intraconnection lines corresponding to said first logical cluster to an adjacent intraconnection line of said first set of intraconnection lines of an adjacent second logical cluster;

controlling a fourth switch for coupling one block connector of said second set of block connectors corresponding to said first logical block to an adjacent block connector of said second set of block connectors of said adjacent second logical block.

36. The method of Claim 35 further comprising the step of controlling fifth switch to couple an output of one of said cells of one of said logical clusters to at least one input of one of said cells of said one of said logical clusters.

37. The method of Claim 35 further comprising the step of selecting whether said second set of block connectors are to conduct input signals or output signals.

Figure 1

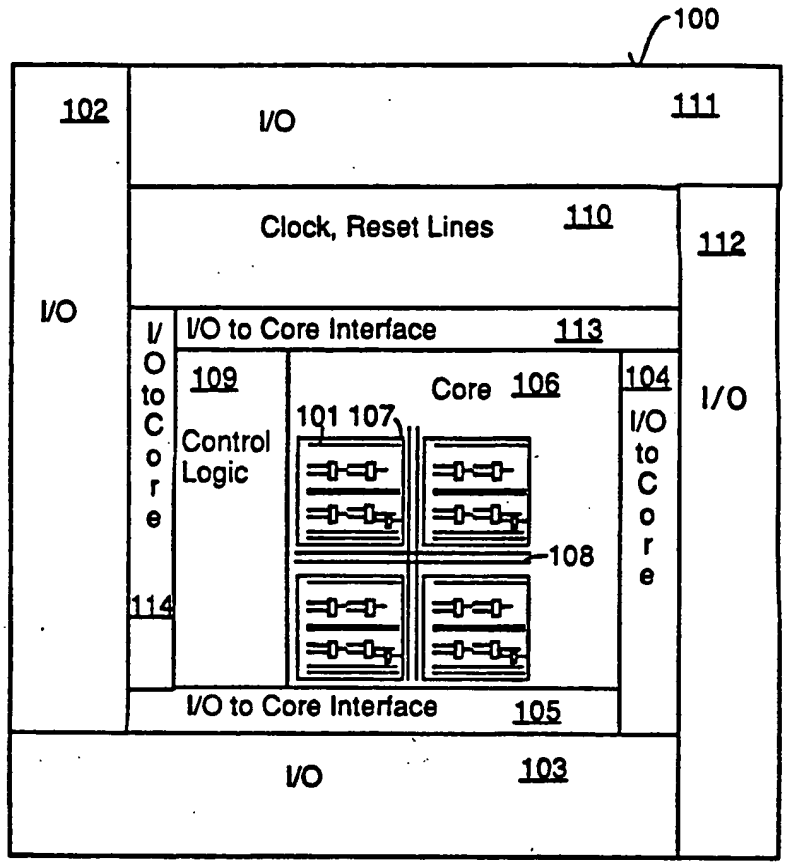

100
102
I/O
111
Clock, Reset Lines
110
112
I/O
I/O to Core Interface
113
I/O to Core
114
109
Control Logic
Core 106
101 107
108
104
I/O to Core
I/O
I/O to Core Interface
105
I/O
103

Figure 2B

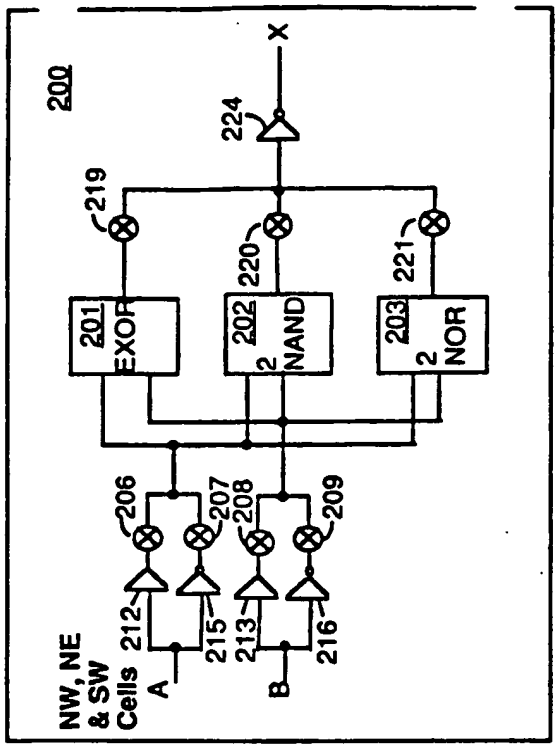


Figure 2A

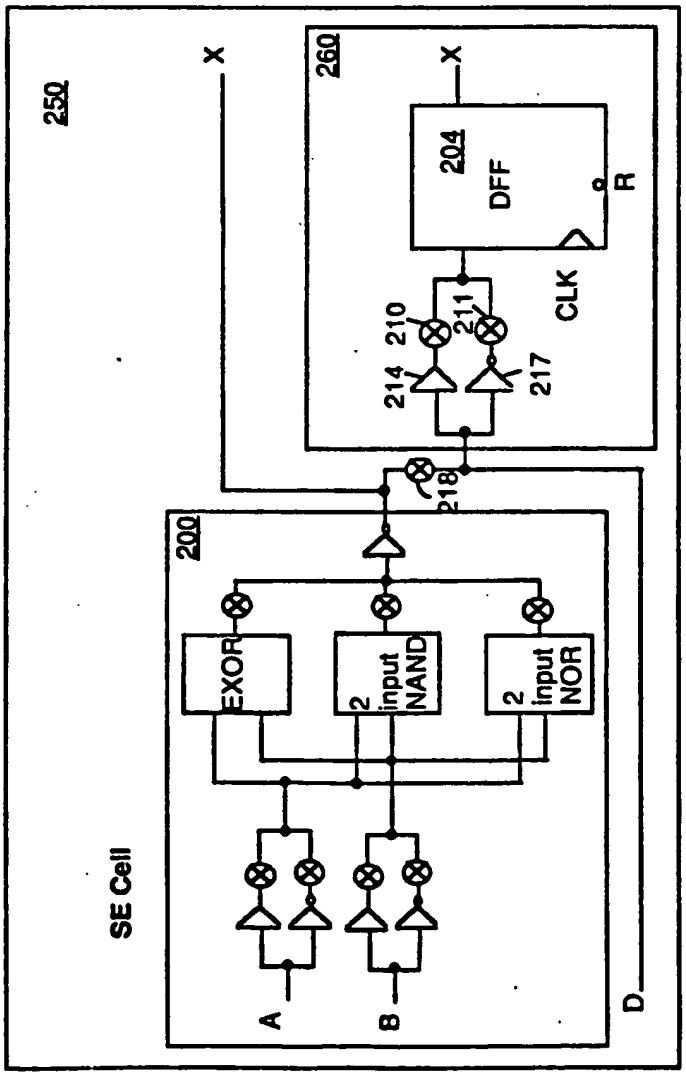


Figure 3A

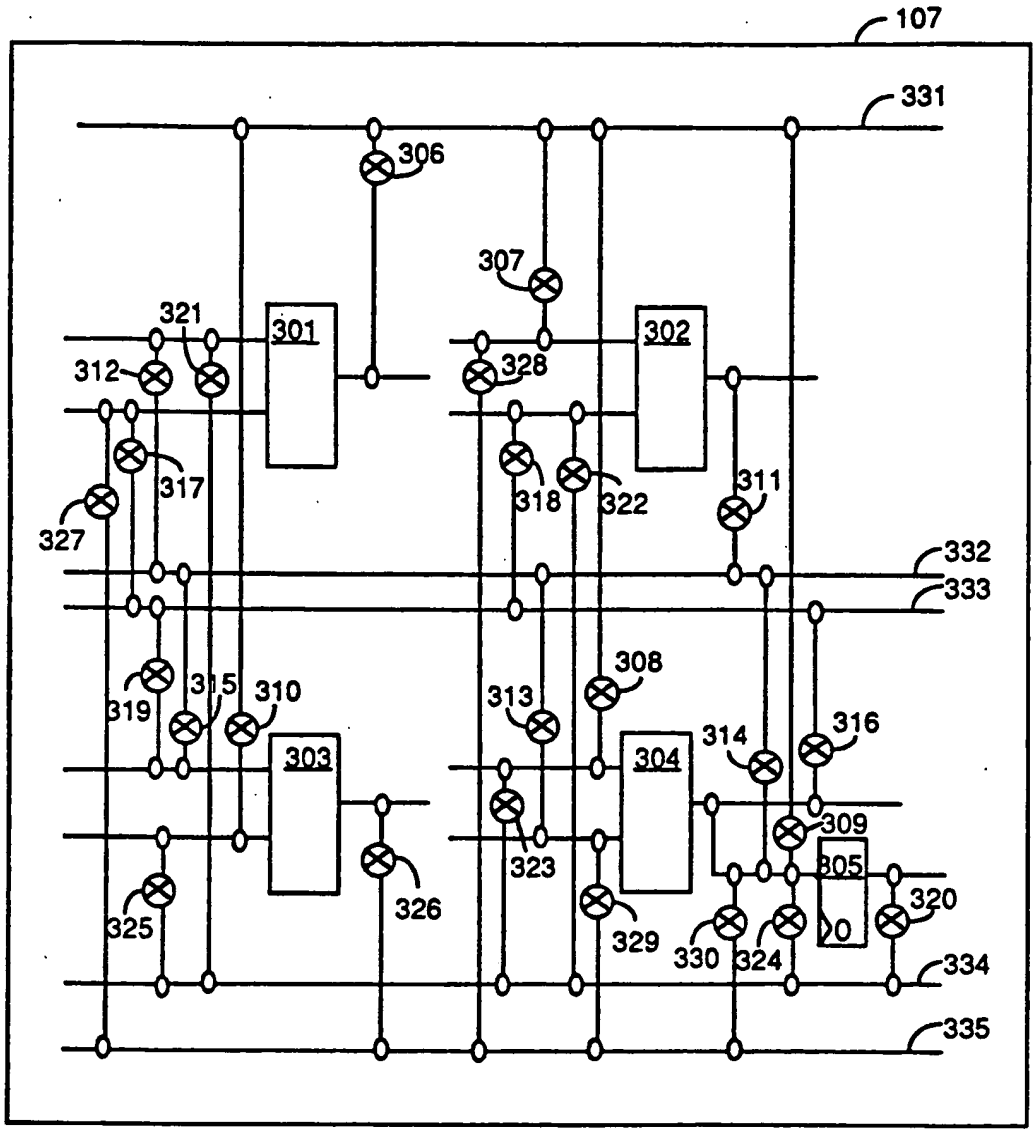

107
331
306
307
321
301
302
312
328
317
318
322
311
327
332
333
319
315
310
308
313
316
303
304
314
309
305
320
323
326
325
329
330
324
334
335

Figure 3B

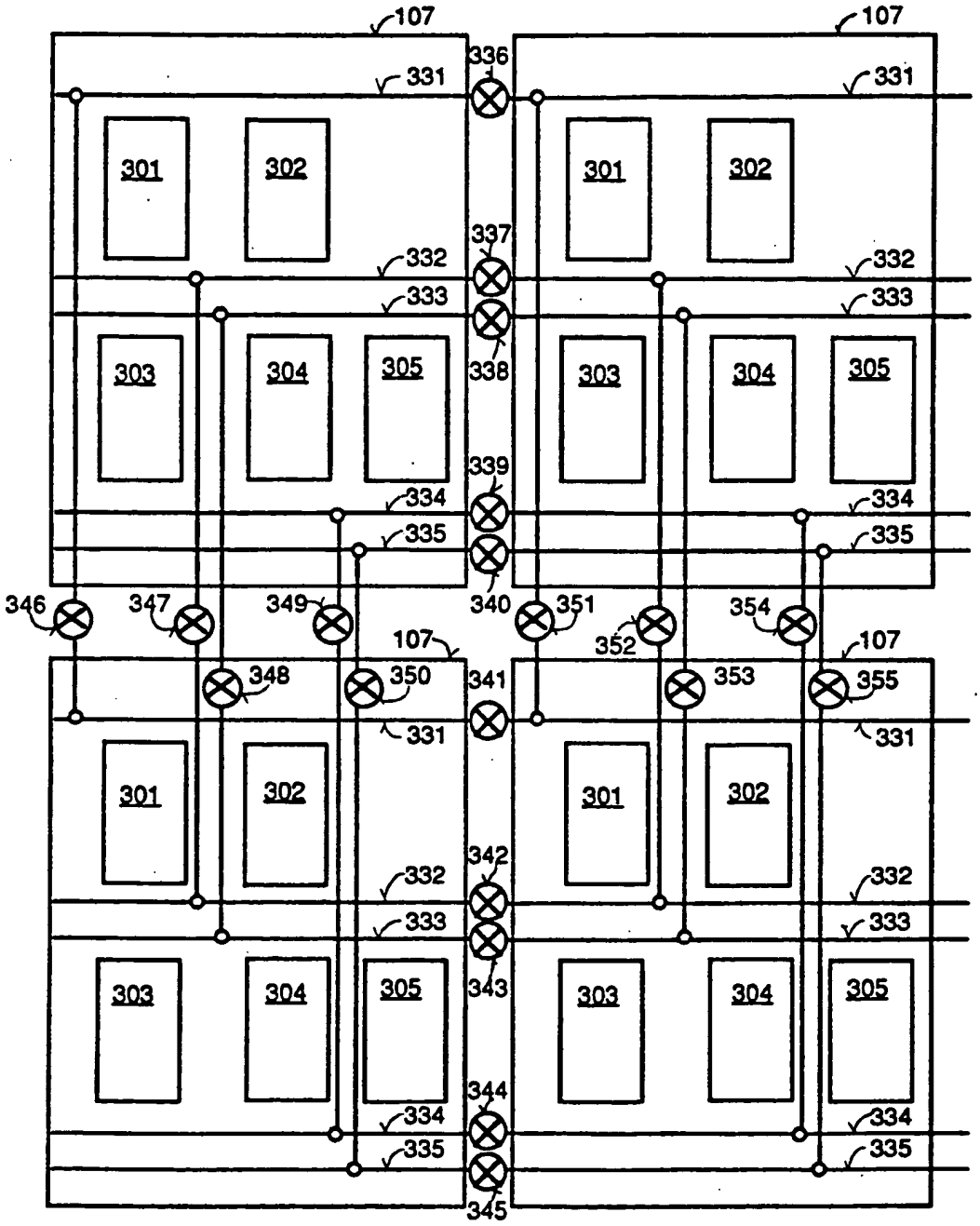

107
331
336
301
302
337
332
333
303
304
305
338
339
334
335
340
346
347
349
351
352
354
348
350
341
353
355
342
343
344
345

Figure 4A

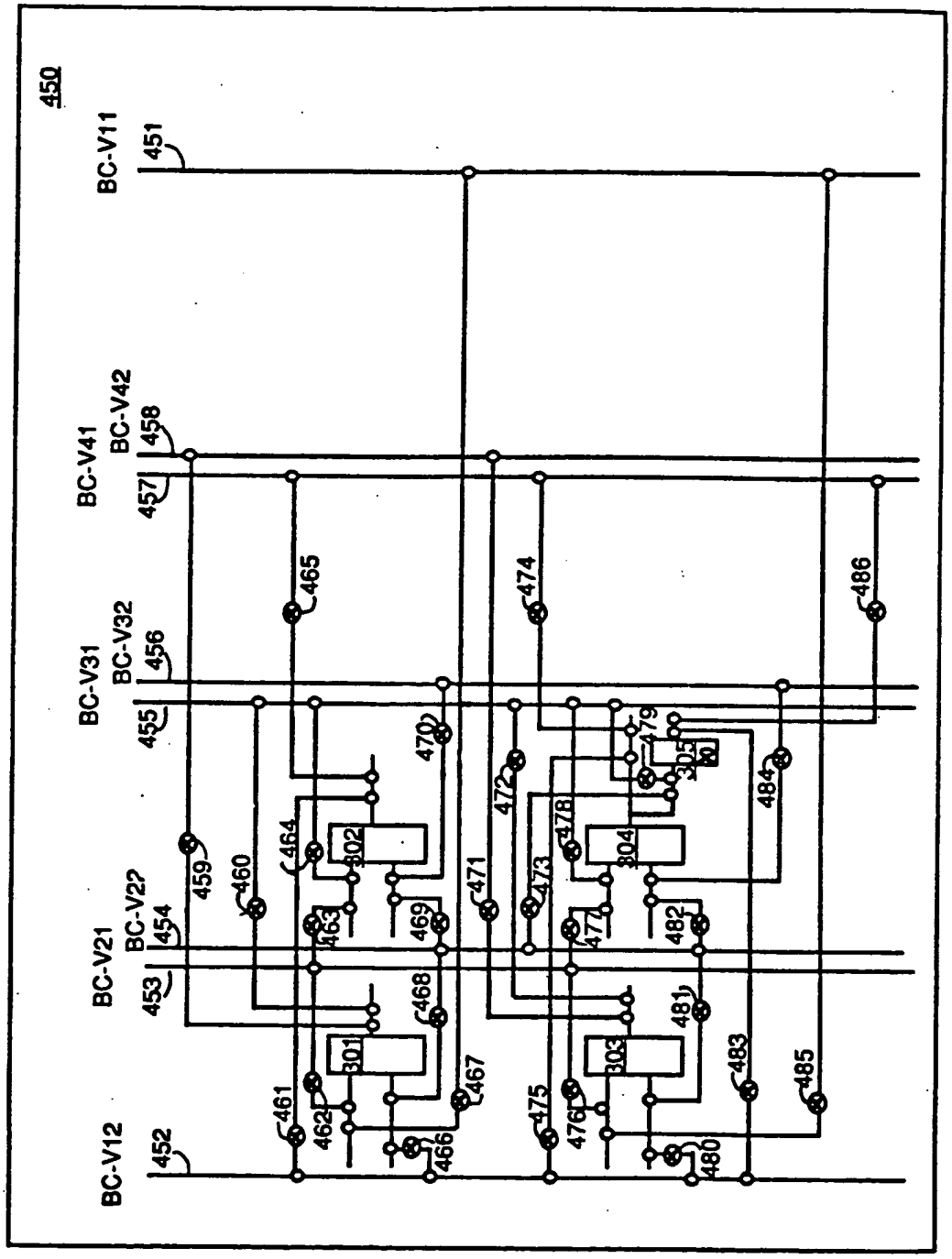

450
BC-V11
451
BC-V42
458
BC-V41
457
BC-V32
456
BC-V31
455
BC-V22
454
BC-V21
453
BC-V12
452
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
301
302
303
304
305

Figure 4B

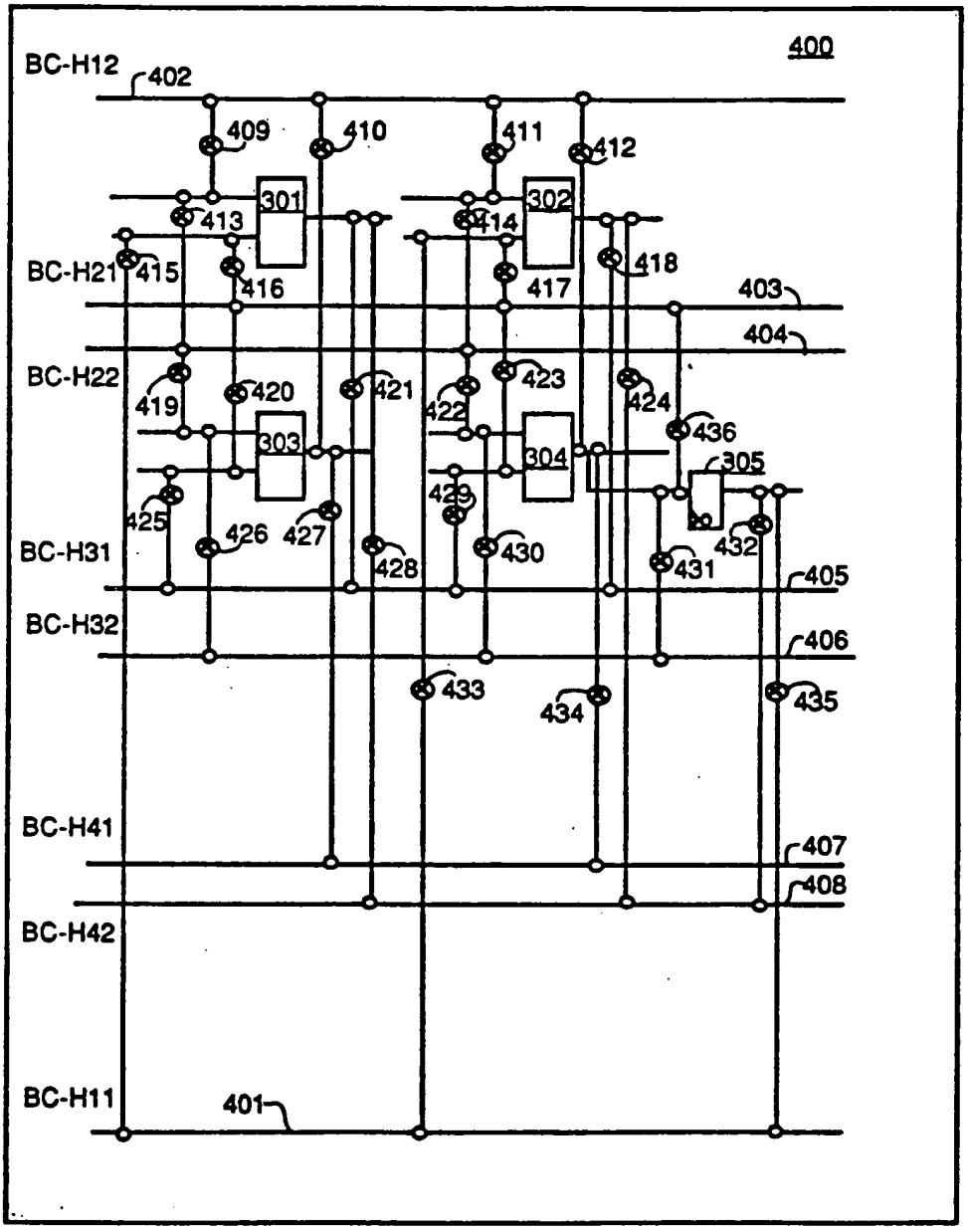

400
BC-H12
402
409
410
411
412
301
302
413
414
BC-H21
415
416
417
418
403
404
BC-H22
419
420
421
422
423
424
436
303
304
305
425
426
427
428
429
430
431
432
BC-H31
405
BC-H32
406
433
434
435
BC-H41
407
408
BC-H42
BC-H11
401

Figure 5A

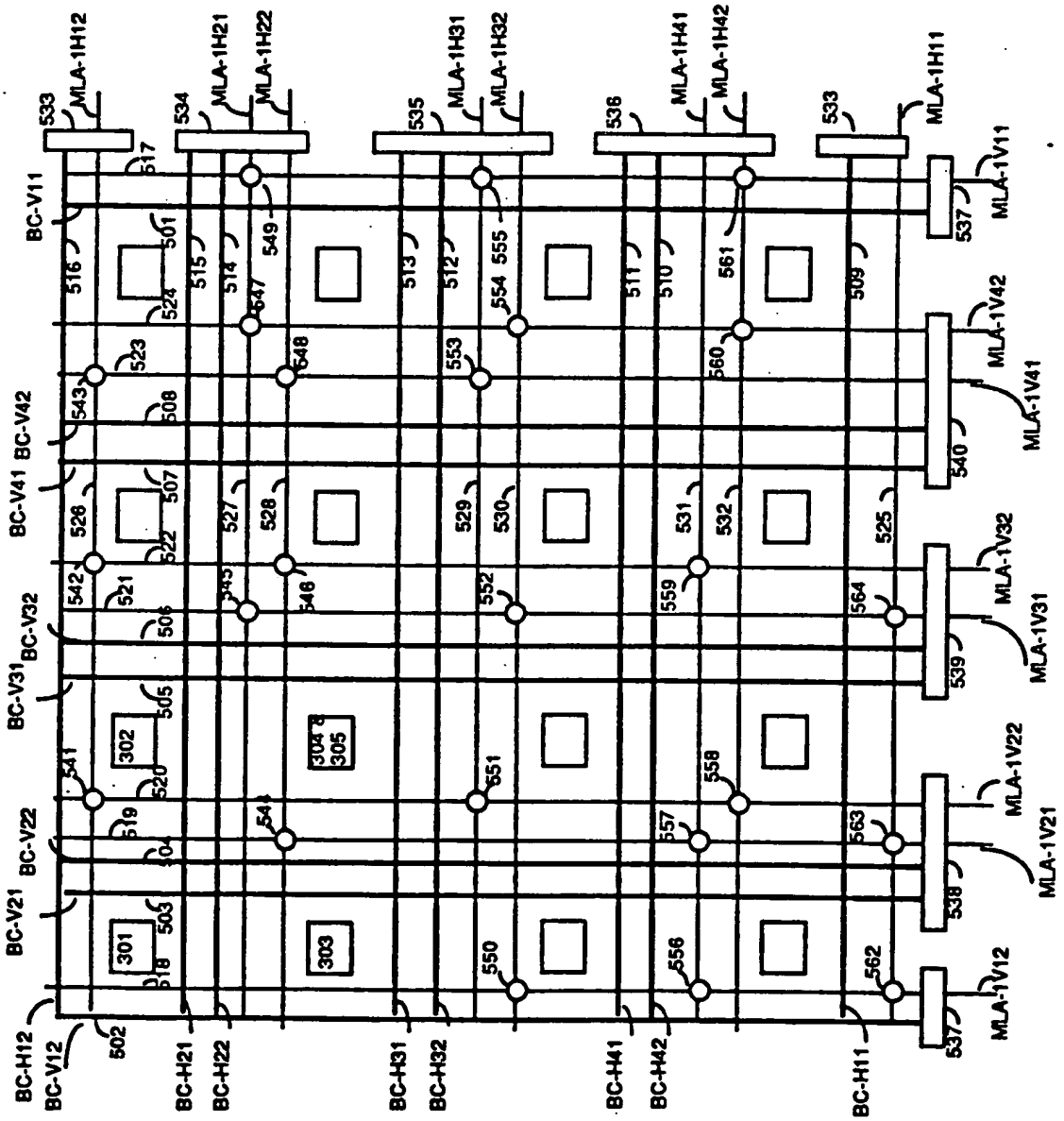

MLA-1H12
MLA-1H21
MLA-1H22
MLA-1H31
MLA-1H32
MLA-1H41
MLA-1H42
MLA-1H11
MLA-1V11
MLA-1V42
MLA-1V41
MLA-1V32
MLA-1V31
MLA-1V22
MLA-1V21
MLA-1V12
BC-V11
BC-V42
BC-V41
BC-V32
BC-V31
BC-V22
BC-V21
BC-V12
BC-H12
BC-H21
BC-H22
BC-H31
BC-H32
BC-H41
BC-H42
BC-H11
301
302
303
304
305

Figure 5C

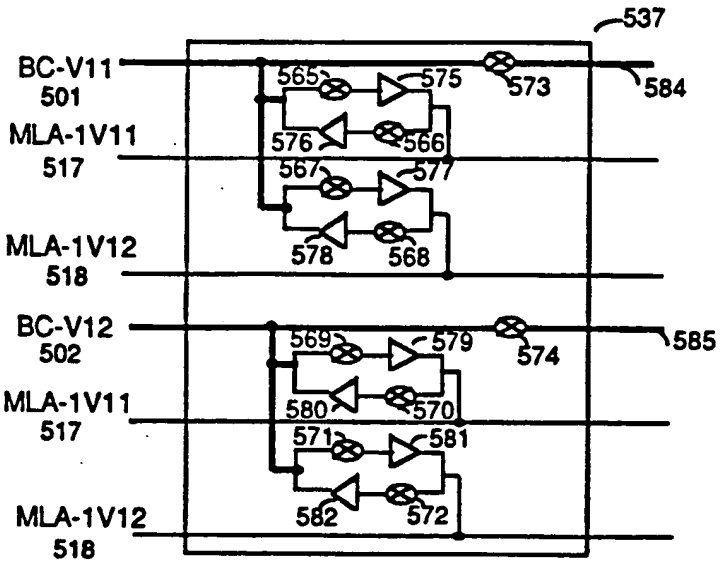


Figure 5B

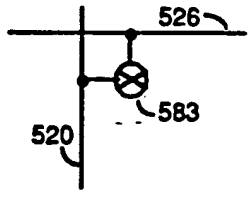


Figure 6

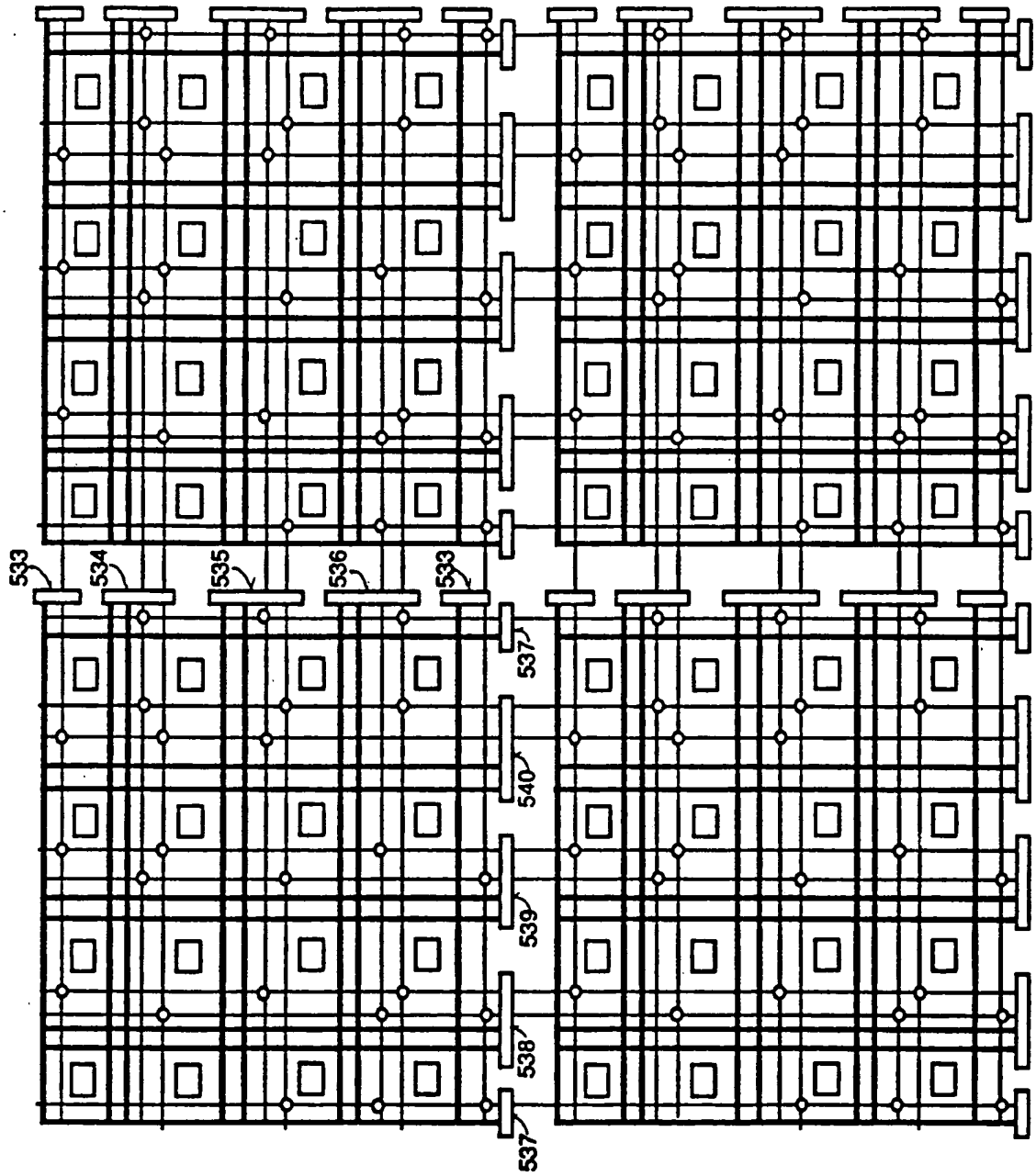


Figure 7B

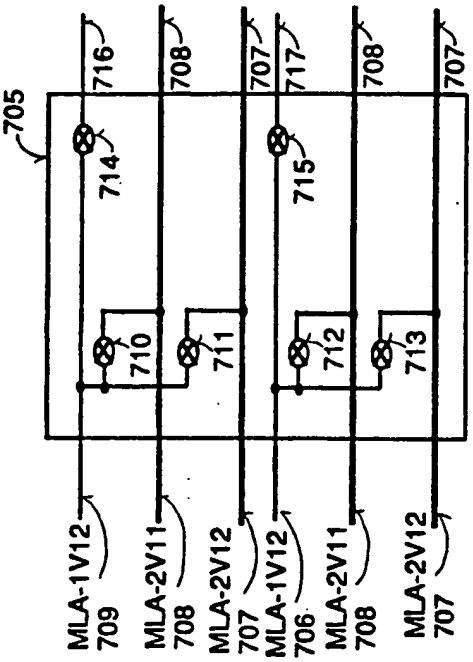


Figure 7A

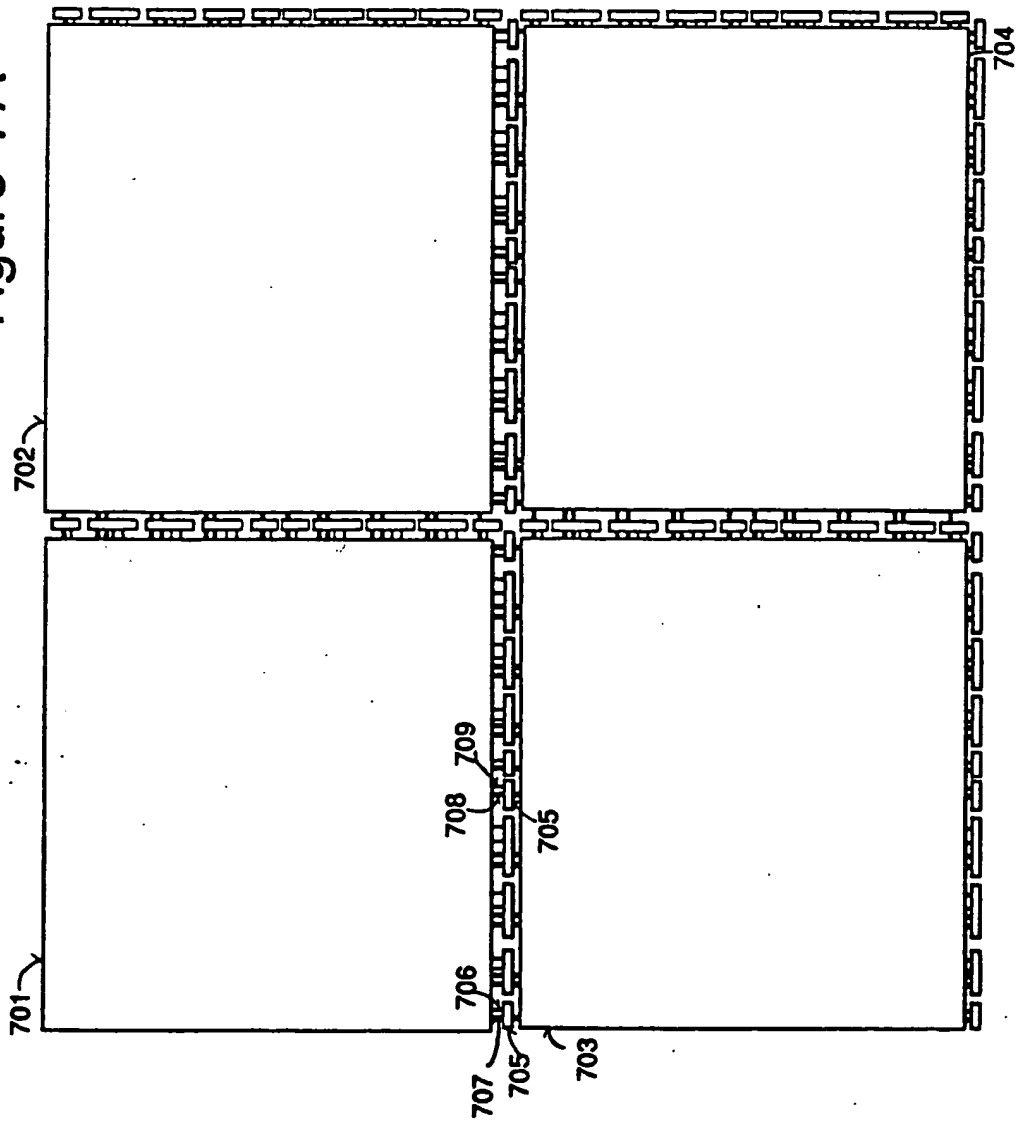


Figure 8A

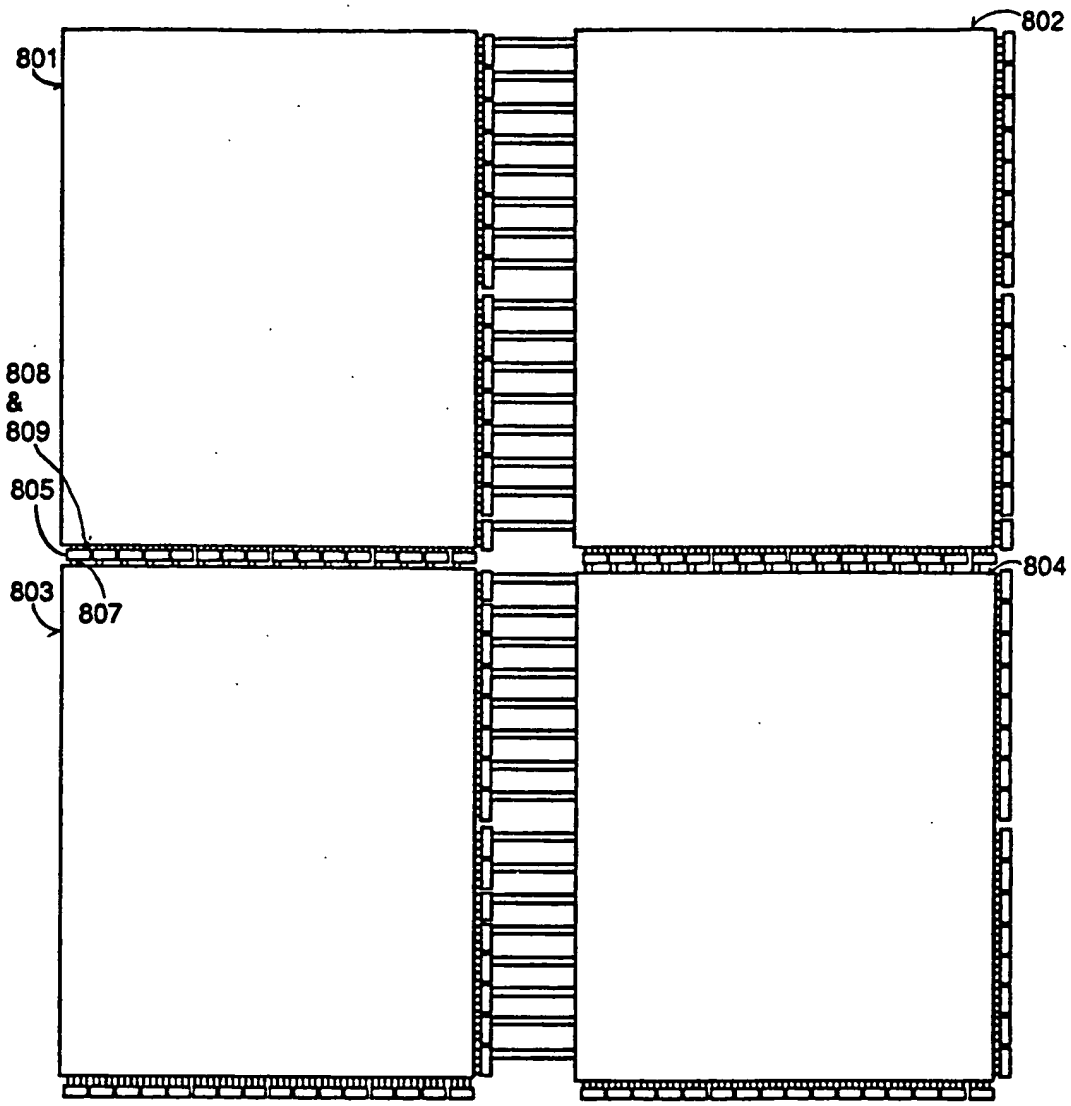
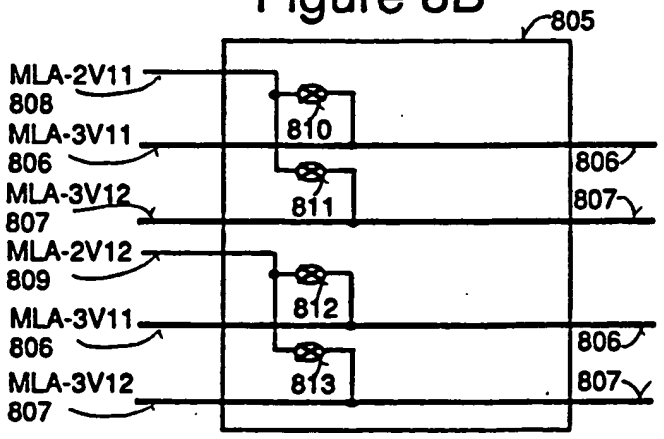


Figure 8B



INTERNATIONAL SEARCH REPORT

Inter nal Application No
PCT/US 94/07187

A. CLASSIFICATION OF SUBJECT MATTER
IPC 6 H03K19/177

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 6 H03K

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO,A,92 08286 (CONCURRENT LOGIC INC.) 14 May 1992 see page P9, line 15 - line 27; figure 3 see page 8, line 9 - page 9, line 14; figures 5,6 see page 10, line 15 - page 12, line 7; figures 7-9	1-3
A	see page 6, line 5 - page 8, line 8; figure 3	4,7
A	see page 10, line 29 - line 30	6
A	see page 15, line 20 - line 35	9,13

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
20 October 1994	31.10.94

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax (+31-70) 340-3016	Feuer, F

Form PCT/ISA/210 (second sheet) (July 1992)

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	PROCEEDINGS OF THE IEEE 1993 CUSTOM INTEGRATED CIRCUITS CONFERENCE, May 1993, NEW YORK pages 7.3.1 - 7.3.5 CLIFF ET AL 'a dual granularity and globally interconnected architecture for a programmable logic device' see figures 1,2	1-3
X	IEEE 1993 CUSTOM INTEGRATED CIRCUITS CONFERENCE, May 1993, NEW YORK (US) pages 7.2.1 - 7.2.5 BRITTON ET AL 'optimized reconfigurable cell array architecture for high-performance field programmable gate arrays' see figures 2-4	1-3
A	EP,A,0 415 542 (ADVANCED MICRO DEVICES) 6 March 1991 see figures 4-15	1-3
A	GB,A,2 180 382 (PILKINGTON MICRO-ELECTRONICS LIMITED) 25 March 1987 see figures 1-7	1-3

1

INTERNATIONAL SEARCH REPORT

.nformation on patent family members

Intc. .onal Application No
PCT/US 94/07187

Patent document cited in search report	Publication date	Patent family member(s)		Publication date
WO-A-9208286	14-05-92	US-A-	5144166	01-09-92
		EP-A-	0555353	18-08-93
		JP-T-	6502523	17-03-94
		US-A-	5245227	14-09-93
		US-A-	5218240	08-06-93
EP-A-0415542	06-03-91	US-A-	5212652	18-05-93
		JP-A-	3078317	03-04-91
		JP-A-	3079125	04-04-91
		JP-A-	3079126	04-04-91
		US-A-	5329460	12-07-94
		US-A-	5233539	03-08-93
		US-A-	5185706	09-02-93
		US-A-	5231588	27-07-93
		US-A-	5255203	19-10-93
GB-A-2180382	25-03-87	AU-B-	593281	08-02-90
		AU-A-	6253086	12-03-87
		CA-A-	1269726	29-05-90
		DE-A-	3630835	09-04-87
		EP-A-	0219221	22-04-87
		FR-A-	2587158	13-03-87
		FR-A-	2593982	07-08-87
		JP-A-	62115844	27-05-87
		US-A-	4935734	19-06-90
		DE-A-	3685629	16-07-92
		EP-A,B	0220816	06-05-87
		GB-A-	2182220	07-05-87
		JP-A-	62124692	05-06-87
		US-A-	4868419	19-09-89